Minutes of a meeting about the use and customization of KOPIO GEANT in Perugia

Present: Riccardo Cenci, Ermanno Imbergamo, Nello Nappi, Marisa Valdata

The purpose of the meeting was:

- review the different uses made of Blecher's Monte-Carlo in Perugia;
- organize the work to standardize the different versions of the program developed for the different studies;
- organize the work to "rephase" the program to the most updated version (all development in Perugia to date has been based on a version frozen on December 2001).

After the meeting was called, we became aware of the development work being organized at BNL. So we also had some discussion on how to coordinate our work with the work being done in Brookhaven.

1. Review of the different uses of the Monte-Carlo made in Perugia

Three studies, at a different level of completion, are being performed:

a) Trigger simulation (Ermanno)
b) Studies of shower geometries (Riccardo)
c) Studies of signal/background for different options of a PR/CAL inner liner (Ermanno+Marisa+Nello)

Each of these studies has used a program branched out of an initially common version. Ermanno has developed the code necessary for use c) in a manner that is compatible with use a). However minor corrections and additions to this code were made by Nello during summer. Thus we have at the moment three separate versions, but they are to a large extent coincident.

We reviewed the main additions to Blecher's code present in these three versions.

One major development by Ermanno, motivated by c), was aimed at integrating fastmc with the GEANT simulation (we call this "mixed" Monte-Carlo), in order to use GEANT treatment only for tracks of interest (in our case, tracks near the beam hole). The way this works is that fastmc is called as kinematics generator for GEANT (in *gukine*) and performs its standard operations including those performed by *SMEAR*. Tracks hitting the preradiator face within a user defined region are stored in a list and, on exit from *SMEAR*, in *gukine*, are put in the GEANT stack. Control is returned to fastmc again by calling *RECON* in *guout*. This is done in order to allow *RECON* to use GEANT results for those particles where they are available.

For studies a) and b) Ermanno and Riccardo have used their own large arrays to store the energies deposited in individual units of the preradiator and calorimeter. In the case of the preradiator, for Ermanno, the unit is a scintillator slab in each layer, for Riccardo is a 20×20 cm^2 tile, again in each layer.

For study c) Nello stores, for each particle, the total (i.e. deposited both in active and passive materials) energy deposited in 4 sets of low level volumes: preradiator, calorimeter, inner liner, all other. Contrary to the standard approach, this requires, in *gustep*, to accumulate the energy deposited also in volumes not declared as "sensitive" to GEANT. This is made in order to use a parameterization for the energy resolution in each detector, rather than relying on GEANT to reproduce it.

2. Discussion on the work needed to standardize the different versions of the program

After some discussion, it was agreed, that, in order to produce a single version suitable for the different usages outlined above, it is necessary to separate clearly the parts that handle the accumulation of energies, that should be standard, from the part where data analyses are performed, which each user will develop in dedicated subroutines. If necessary, different running options can be defined, in order to customize also the first part. For example, the code needed to store the energy stored in passive and active materials (for study c), should be included in the standard code, but in a way that it can be enabled or disabled on the basis of a flag read from title files. Also the output should be made standard. Ermanno proposed to use a technique similar to the NA48 "ntmaker" to define nt-ples composed of individual sections that could be enabled or disabled by user defined flags.

There was, then, some discussion about which energies should be accumulated and with which granularity. The accumulation in tiles for the preradiator requires large memory in the nt-ples and does not look necessary, unless realistic proposals for a tile readout pop out in the collaboration. So we decided that only the following information should be recorded:

- energy deposited in units corresponding to each single photomultiplier as foreseen in the detector design; for the preradiator this would imply, for example, summing the energy deposited in corresponding strips with the same orientation inside one module;
- energy deposited by each particle in the preradiator volume, the calorimeter volume, in 40 inner liner elements, and in all other volumes, including active and passive materials.

In order to save nt-ple volume we should store information only for those units with an energy different from zero (unless we find out that a fraction of channels close to ½ satisfies this condition).

Another point that was raised is the fact that the standard version of the program that we have used does not support a subdivision of the volumes in units corresponding to individual readout channels. The subdivision is done "by hand" in *gustep* using the track coordinates. If this can be agreed with the U.S. collaborators, we feel that it would be much more transparent, to implement the subdivision of the volumes (if necessary on the basis of parameters defined in "title" files) in the definition of the geometry and make use of the utilities provided by the GEANT "detector response package" (described in section HITS of the manual) to associate volumes to readout channels.

3. Discussion on the work needed to "rephase" the program to the most updated version

Due to the large amount of development work which has been performed in a non-coordinated way since last December, it appears that there is no alternative to the following painful procedure:

- download the latest version of the standard program;
- go through the changes required for our local studies and re-implement them using the previous work as a guide-line, but following the new policies discussed in point 2.

However, in order to avoid to run again into similar problems, this work should be coordinated with the U.S. collaborators that are currently working on the development of the program. We should discuss with them our special requirements outlined in point 2 in order to understand if they can be embedded in the standard version of the program or if they should be developed as special versions. Even in this second case, proper hook-ups to the non-standard codes should be provided in the standard program version. We should also discuss with them if it could be useful to isolate parts of the development work that could be performed in Perugia.

Whether or not persons from the Perugia group join the group of developers, it is important that we agree with the U.S. collaborators on a set of rules for bug corrections and new version releases, in order to minimize the risk of clashes occurring when several persons are working on the same program.